1. Giới thiệu
   1. Giới thiệu đề tài
   2. Mục tiêu và phạm vi đề bài
2. Phân tích vấn đề và hướng giải quyết
   1. Nhận diện giọng nói
      1. Vấn đề

Nhận dạng giọng nói là một quá trình nhận dạng mẫu, với mục đích là phân lớp (classify) thông tin đầu vào là tín hiệu tiếng nói thành một dãy tuần tự các mẫu đã được học trước đó và lưu trữ trong bộ nhớ. Các mẫu là các đơn vị nhận dạng, chúng có thể là các từ, hoặc các âm vị. Nếu các mẫu này là bất biến và không thay đổi thì công việc nhận dạng giọng nói trở nên đơn giản bằng cách so sánh dữ liệu giọng nói cần nhận dạng với các mẫu đã được học và lưu trữ trong bộ nhớ. Khó khăn cơ bản của nhận dạng giọng nói đó là giọng nói luôn biến thiên theo thời gian và có sự khác biệt lớn giữa giọng nói của những người nói khác nhau, tốc độ nói, ngữ cảnh và môi trường âm học khác nhau. Xác định những thông tin biến thiên nào của giọng nói là có ích và những thông tin nào là không có ích đối với nhận dạng giọng nói là rất quan trọng.

Thoạt nhìn thì việc triển khai công nghệ nhận dạng giọng nói khá đơn giản, nhưng thực chất thì không phải như thế. Thứ nhất, chúng ta cần phải xây dựng nên một công nghệ có thể lắng nghe, phân tích và phiên dịch một cách chính xác giọng nói của người dùng. Công nghệ này yêu cầu độ chính xác cao, để thiết bị có thể hiểu và phiên dịch chính xác những gì người dung nói, tránh gây ức chế cho người sử dụng. Có một kĩ thuật được nhắc đến nhiều trong thời gian gần đây, đó là Xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP). Nó là tập hợp của nhiều thuận toán phức tạp nhằm phân tích mệnh lệnh của người dùng nhưng không bắt buộc họ phải nói theo một cấu trúc câu định sẵn. Tuy nhiên NLP cũng không phải là đơn giản để phát triển. Cả Apple, Google và Microsoft đều phải đầu tư rất nhiều tiền bạc và nguồn lực để có thể đưa NLP lên đến mức tiến bộ như hiện nay. Mặc dù vậy, các công ty vẫn phải tiếp tục nghiên cứu để cải thiện độ chính xác.

Bên cạnh đó, vấn đề bản địa hóa (localization) cũng là một chuyện làm đau đầu các lập trình viên. Mỗi quốc gia sẽ có ngôn ngữ của riêng mình, vấn đề đó là làm thế nào để có thể hỗ trợ càng nhiều ngôn ngữ càng tốt.

* + 1. Giải pháp

Bởi vì việc nghiên cứu về nhận dạng giọng nói cũng như công nghệ xử lý ngôn ngữ tự nhiên đòi hỏi việc đầu tư nhiều về thời gian và nguồn lực nên nhóm quyết định sử dụng các thư viện và các api có sẵn của Google, Apple, Microsoft để phục vụ cho việc nhận dạng giọng nói trong ứng dụng.

* 1. Độ chính xác của các api
     1. Vấn đề

Công nghệ nhận diện giọng nói hiện nay của Google, Apple, Microsoft rất phát triển. Tuy nhiên, hiện nay công nghệ giọng nói vẫn chưa thể thay thế hoàn toàn bàn phím ảo/vật lý hoặc các nút trên màn hình. Các công ty vẫn phải tiếp tục nghiên cứu để cải thiện độ chính xác. Vì thế, vấn đề về độ chính xác của các api mà nhóm sử dụng cũng được đặt ra : Cần xử lý các đoạn văn bản được trả về từ các api.

* + 1. Giải pháp

Với vấn đề trên, nhóm quyết định xử dụng thêm các thư viện xử lý ngôn ngữ tự nhiên hiện có để xử lý các đoạn văn bản trả về từ api cho phù hợp với cấu trúc của một bài báo cáo. Nghĩa là sử dụng thư viện để xử lý đoạn văn bản bao gồm tiêu đề của từng trường và dữ liệu trả về tương ứng.

Bên cạnh đó, ứng dụng cũng cần cho phép người sử dụng chỉnh sửa những lỗi sai bằng bàn phím ảo của các thiết bị di động.

* 1. Âm điệu, âm vực khác nhau.
     1. Vấn đề

Giọng nói của mỗi người có âm điệu, âm vực khác nhau và biến thiên theo thời gian, không gian nên việc nhận dạng giọng nói không đơn thuần là việc so trùng các mẫu lưu trữ với các mẫu đầu vào.

* + 1. Giải pháp

Các api nhận diện giọng nói hiện nay có khả năng phân biệt âm điệu, âm vực khá ổn nên nhóm chọn dựa vào api để giải quyết vấn đề này.

* 1. Xử lí tiếng ồn của dữ liệu đầu vào
     1. Vấn đề

Trong hiện thực, người sử dụng dùng các thiết bị di động để báo cáo nhanh trong môi trường có nhiều tiếng ồn. Việc này sẽ gây ảnh hưởng tới việc nhập dữ liệu đầu vào bằng giọng nói nên âm thanh đầu vào cần được xử lý lọc tạp âm để tăng độ chính xác cho ứng dụng.

* + 1. Giải pháp

Các api nhận diện giọng nói hiện nay có khả năng xử lý tiếng ồn nên nhóm chọn dựa vào api để giải quyết vấn đề này.

* 1. Quản lý dữ liệu, người dùng
     1. Vấn đề

Ứng dụng cần là mối liên kết giữa nhân viên và cấp trên trong việc báo cáo kết quả của các cuộc họp. Vì thế, dữ liệu về các bản báo cáo cần được quản lý rõ ràng. Bên cạnh đó, các bản báo cáo cần được chuyển đến cấp trên ngay khi nó được tạo xong để được kiểm duyệt. Cùng với đó, cấp trên cũng cần biết được người tạo báo cáo, thời gian tạo và địa điểm để kiểm tra, quản lý nhân viên của mình.

* + 1. Giải pháp

Với vấn đề về việc quản lý, nhóm quyết định sử dụng server firebase để lưu trữ dữ liệu. Ngoài ra, bên cạnh ứng dụng di động dành cho nhân viên để tạo báo cáo, nhóm sẽ xây dựng thêm một trang web dành cho người quản lý để quản lý các bản báo cáo, nhân viên, thống kê các thông số. Trang web và ứng dụng sẽ sử dụng cùng một server để đảm bảo sự đồng bộ dữ liệu của các bản báo cáo. Ứng dụng sẽ yêu cầu thiết bị kết nối với mạng để gửi các bản báo cáo lên server. Cùng với đó, các bản báo cáo đều có yêu cầu về người tạo, thời gian và địa điểm tạo. Những thông số này sẽ được tạo cách tự động khi báo cáo được tạo.

* 1. Ngôn ngữ hỗ trợ

1. Vấn đề:

Vấn đề bản địa hóa cho từng quốc gia là một vấn đề rất quan trọng đối với việc phát triển các ứng dụng nhận dạng giọng nói. Vì mỗi quốc gia đều có ngôn ngữ riêng của mình, mỗi ngôn ngữ đều có đặc thù âm vực và cú pháp câu nói khác nhau, chưa nói đến việc, mỗi vùng miền trên một quốc gia lại có cách nói, giọng điệu khác nhau mặc dù họ sử dụng chung một ngôn ngữ. Hiện nay hầu hết các thư viện và api hỗ trợ nhận dạng giọng nói đều hỗ trợ các ngôn ngữ phổ biến như tiếng Anh, tiếng Pháp, tiếng Nhật,... Tuy nhiên các ngôn ngữ ít phổ biến cụ thể là tiếng Việt thì không được phần lớn các api hỗ trợ hoặc có hỗ trợ nhưng còn nhiều hạn chế.

1. Giải pháp:

Về vấn đề ngôn ngữ hỗ trợ, dựa vào tính phổ biến của ngôn ngữ, nhóm quyết định chọn tiếng Anh, tiếng Nhật là hai ngôn ngữ được hỗ trợ. Ngoài ra, trong thời gian hiện thực ứng dụng, nếu nhóm tìm ra được thư viện hỗ trợ đầy đủ ngôn ngữ tiếng Việt thì nhóm sẽ bổ sung tiếng Việt vào nhóm ngôn ngữ được hỗ trợ.

* 1. Lưu trữ báo cáo

1. Vấn đề:

Các ứng dụng hoạt động trên nền tảng mobile đều bị hạn chế về không gian lưu trữ và tốc độ xử lý dữ liệu. Trong điều kiện thực tế, mỗi nhân viên khi làm việc đều phải thực hiện rất nhiều các loại báo cáo khác nhau. Nếu lưu trữ báo cáo tại bộ nhớ của điện thoại, lâu dài sẽ chiếm nhiều tài nguyên của điện thoại. Bên cạnh đó, với điều kiện thực tế là nhiều nhân viên sử dụng ứng dụng cùng lúc, việc lưu trữ dữ liệu phía client sẽ gây khó khăn cho việc đồng bộ và quản lý báo cáo của cấp trên.

1. Giải pháp:

Để giải quyết vấn đề về về lưu trữ báo cáo, nhóm quyết định dùng service Firebase để lưu trữ dữ liệu. Tất cả các dữ liệu báo cáo của nhân viên sẽ được lưu trữ chung trên Firebase, việc lưu trữ các báo cáo ở client chỉ là tạm thời và nếu vượt quá một mức giới hạn nhất định thì dữ liệu sẽ bị xóa để tranh việc chiếm nhiều tài nguyên của điện thoại.

* 1. Firebase Database:
* Firebase là một dịch vụ cơ sở dữ liệu thời gian thực hoạt động trên nền tảng đám mây được cung cấp bởi Google nhằm giúp các lập trình phát triển nhanh các ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu.
* Firebase lưu trữ dữ liệu database dưới dạng JSON và thực hiện đồng bộ database tới tất cả các client theo thời gian thực. Cụ thể hơn là bạn có thể xây dựng được client đa nền tảng (cross-platform client) và tất cả các client này sẽ cùng sử dụng chung 1 database đến từ Firebase và có thể tự động cập nhật mỗi khi dữ liệu trong database được thêm mới hoặc sửa đổi. Ngoài ra Firebase còn cho phép bạn phân quyền một các đơn giản bằng cú pháp tương tự như javascript.
* Lợi ích từ việc sử dụng Google Firebase:
  + Triển khai ứng dụng cực nhanh: Sử dụng Firebase giúp giảm bớt rất nhiều thời gian cho việc viết các dòng code để quản lý và đồng bộ cơ sở dữ liệu, mọi việc sẽ diễn ra hoàn toàn tự động với các API của Firebase. Không chỉ có vậy Firebase còn hỗ trợ đa nền tảng nên sẽ đỡ mất thời gian rất nhiều khi ứng dụng bạn muốn xây dựng là ứng dụng đa nền tảng.
  + Bảo mật: Firebase hoạt động dựa trên nền tảng cloud và thực hiện kết nối thông qua giao thức bảo mật SSL, chính vì vậy bạn sẽ bớt lo lắng rất nhiều về việc bảo mật của dữ liệu cũng như đường truyền giữa client và server. Không chỉ có vậy, việc cho phép phân quyền người dùng database bằng cú pháp javascipt cũng nâng cao hơn nhiều độ bảo mật cho ứng dụng.
  + Tính linh hoạt và khả năng mở rộng: Sử dụng Firebase sẽ giúp bạn dễ dàng hơn rất nhiều mỗi khi cần nâng cấp hay mở rộng dịch vụ. Ngoài ra firebase còn cho phép bạn tự xây dựng server của riêng mình để bạn có thể thuận tiện hơn trong quá trình quản lý.
  + Sự ổn định: Firebase hoạt động dựa trên nền tảng cloud đến từ Google vì vậy hầu như bạn không bao giờ phải lo lắng về việc sập server, tấn công mạng như DDOS, tốc độ kết nối… Bởi đơn giản là Firebase hoạt động trên hệ thống server của Google. Hơn nữa nhờ hoạt động trên nền tảng Cloud nên việc nâng cấp, bảo trì server cũng diễn ra rất đơn giản mà không cần phải dừng server để nâng cấp như truyền thống.
  1. Nhận diện dấu chấm, dấu phẩy, chữ số
     1. Vấn đề:

Trong thực tế, để tạo một báo cáo bằng giọng nói giống với thực tế, thì báo cáo yêu cầu phải có các dấu chấm câu, chữ số rõ ràng như dấu “.”, dấu “,”, “1”, “100”.... Việc phân biệt giữa câu lệnh dấu “.” và nội dung “dấu chấm” không phải là việc dễ dàng.

* + 1. Giải pháp:

Dấu chấm, phẩy:

Ngoài việc sử dụng api hỗ trợ tốt việc nhận dạng dấu chấm, dấu phẩy, nhóm quyết định nâng cao khả năng nhận diện dựa vào việc phân tích khoảng lặng khi nhập giọng nói từ người dùng.

Chữ số:

Để nhận dạng giọng nói, nhóm quyết định sẽ đặt ra các cú pháp mẫu, nếu trong văn bản có đoạn text giống cú pháp mẫu thì sẽ chuyển chữ thành số.

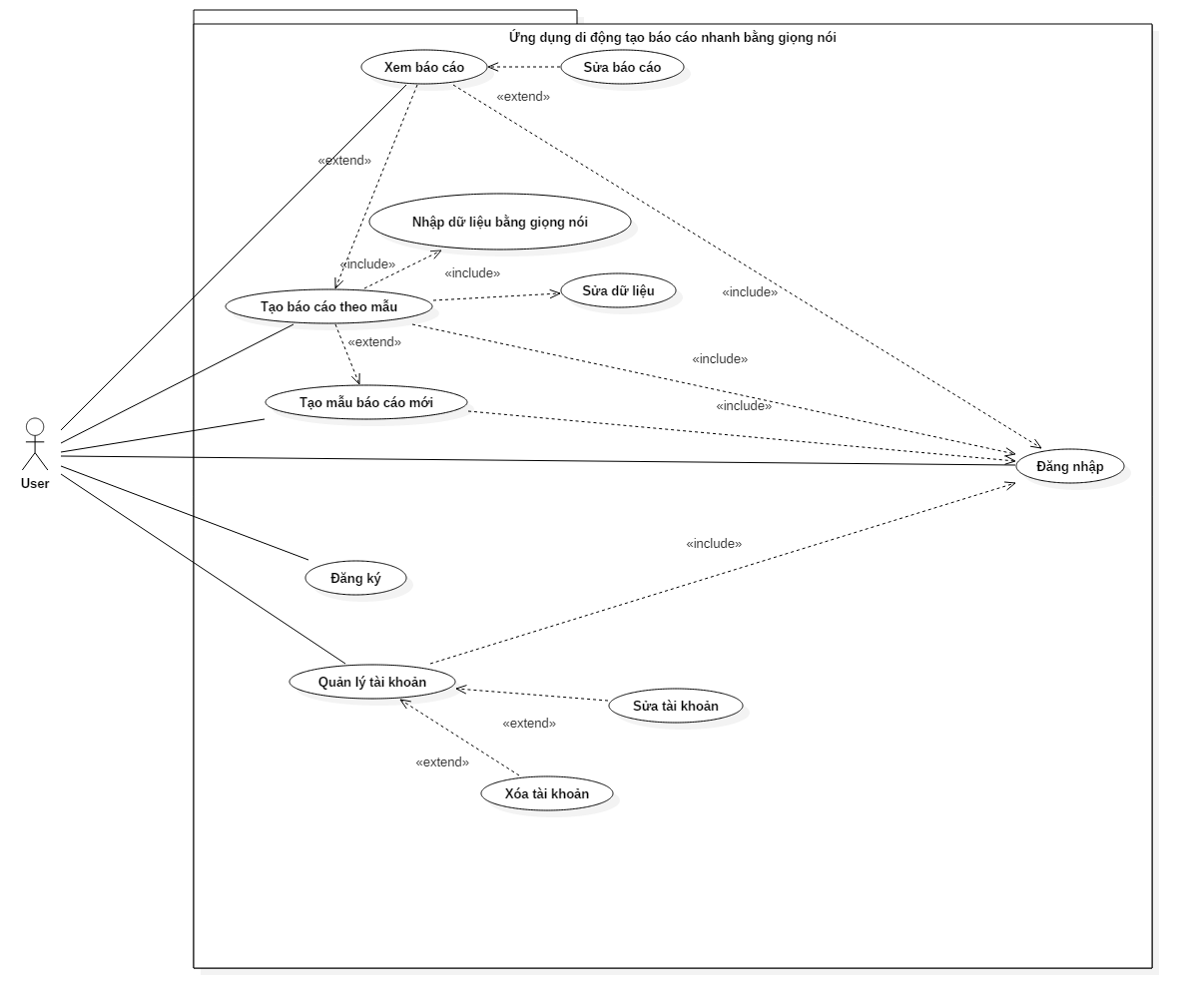
Ngoài ra, ứng dụng sẽ cho phép người dùng sửa thủ công bằng bàn phím ảo/vật lý.

* 1. Các thông số quan tâm
  2. Bảo mật thông tin
  3. Lưu trữ dữ liệu
  4. Kết nối mạng

1. Các ứng dụng, công nghệ hiện tại
   1. Google Speech To Text API
   2. IBM Watson Speech To Text API
   3. Microsoft Speech API
   4. Một số ứng dụng sử dụng công nghệ Speech To Text
2. Phân tích hệ thống
   1. Chức năng hệ thống
      1. Người dùng
      2. Người quản trị
   2. Use Case Diagram
      1. Ứng dụng di động tạo báo cáo nhanh bằng giọng nói
         1. Overview

Ứng dụng giúp tạo báo nhanh bằng giọng nói cho nhân viên. Nhân viên có thể tạo báo cáo theo mẫu có sẵn hoặc tạo mẫu báo cáo mới. Thông qua ứng dụng, nhân viên có thể nộp báo cáo nhanh chóng mà không cần trực tiếp gặp mặt người quản lý.

* + - 1. Usecase diagram



|  |  |  |  |
| --- | --- | --- | --- |
| No | Use case ID | Use case name | Description |
| 1 1 | UC\_APP\_01 | Đăng nhập | Cho phép người dùng đăng nhập vào hệ thống |
| 2 | UC\_APP\_02 | Đăng ký | Cho phép người dùng đăng ký tài khoản |
| 3 | UC\_APP\_03 | Quản lý tài khoản | Cho phép người dùng quản lý tài khoản |
| 4 | UC\_APP\_03\_01 | Sửa tài khoản | Cho phép người dùng sửa thông tin tài khoản của mình |
| 5 | UC\_APP\_03\_02 | Xóa tài khoản | Cho phép người dùng xóa tài khoản của mình |
| 6 | UC\_APP\_04 | Tạo báo cáo theo mẫu | Cho phép người dùng tạo báo cáo mới bằng các mẫu báo cáo có sẵn |
| 7 | UC\_APP\_04\_01 | Nhập dữ liệu bằng giọng nói | Cho phép người dụng nhập dữ liệu đầu vào thông qua giọng nói |
| 8 | UC\_APP\_04\_02 | Sửa dữ liệu | Cho phép người dụng sửa dữ liệu sau từng bước nhập dữ liệu bằng giọng nói |
| 9 | UC\_APP\_05 | Tạo mẫu báo cáo mới | Cho phép người dùng tạo và lưu trữ mẫu báo cáo mới |
| 10 | UC\_APP\_06 | Xem báo cáo | Cho phép người dùng xem các báo cáo đã lưu |
| 11 | UC\_APP\_06\_01 | Sửa báo cáo | Cho phép người dùng sửa các báo cáo đã lưu |

* + - 1. Đặc tả usecase

|  |  |
| --- | --- |
| ID | UC\_APP\_01 |
| Name | Đăng nhập |
| Summary | Chức năng đăng nhập vào ứng dụng |
| Actors | User |
| Precondition | User có tài khoản hợp lệ |
| Trigger | * Người dùng truy cập vào ứng dụng * Hệ thống yêu cầu nhập username và password * Người dùng nhập username và password * Hệ thống kiểm tra username và password |
| Basic flow | * Username và password hợp lệ * Hệ thống cung cấp các quyền tương ứng và hiển thị đăng nhập thành công |
| Alternative flow | Người dùng không thể đăng nhập vào hệ thống và nhận thông báo :   * Chưa kích hoạt tài khoản * Tài khoản không hợp lệ * Tài khoản không tồn tại * Nhập sai password * Hệ thống tạm thời bảo trì |
| Post conditions | User đăng nhập thành công |

|  |  |
| --- | --- |
| ID | UC\_APP\_02 |
| Name | Đăng ký |
| Summary | Chức năng đăng ký tài khoản để truy cập ứng dụng |
| Actors | User |
| Precondition | User chưa có tài khoản và muốn truy cập vào hệ thống |
| Trigger | * Người dùng truy cập vào ứng dụng * Hệ thống yêu cầu nhập username và password * Người dùng bấm vào button “Đăng ký” * Người dùng nhập email, username và password * Người dùng chọn “Tạo tài khoản” * Hệ thống gửi mã số kích hoạt tới email đăng ký * Người dùng nhập mã số kích hoạt * Người dùng chọn “Xác nhận” |
| Basic flow | * Email, username và password hợp lệ * Mã số kích hoạt hợp lệ * Hiển thị đăng ký thành công tài khoản * Đăng nhập thành công bằng tài khoản vừa đăng ký |
| Alternative flow | Người dùng không thể đăng ký tài khoản thành công và nhận thông báo :   * Username đã tồn tại * Username hoặc password không hợp lệ * Mã xác nhận không hợp lệ * Hệ thống tạm thời bảo trì |
| Post conditions | User đăng ký tài khoản thành công |

|  |  |
| --- | --- |
| ID | UC\_ APP \_03 |
| Name | Quản lý tài khoản |
| Summary | Chức năng quản lý tài khoản của người dùng |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ |
| Trigger | Người dùng sử dụng chức năng quản lý tài khoản của hệ thống |
| Basic flow | * Hệ thống hiển thị thông tin tài khoản * Người dùng có thể sử dụng các chức năng sửa, xóa tài khoản |
| Post conditions | Hiển thị thông tin tài khoản cho phép người dùng sử dụng các chức năng tương ứng |

|  |  |
| --- | --- |
| ID | UC\_APP\_03\_01 |
| Name | Sửa tài khoản |
| Summary | Chức năng sửa thông tin của tài khoản của người dùng |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ và sử dụng chức năng quản lý tài khoản của hệ thống |
| Trigger | * Người dùng sử dụng chức năng sửa tài khoản |
| Basic flow | * Hệ thống hiển thị thông tin của tài khoản ở dạng có thể sửa đổi * User có thể sửa một số thông tin của tài khoản * User chọn lưu thông tin |
| Post conditions | Lưu thay đổi của thông tin tài khoản của user vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| ID | UC\_APP\_03\_02 |
| Name | Xóa tài khoản |
| Summary | Chức năng xóa tài khoản của người dùng |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ và sử dụng chức năng quản lý tài khoản của hệ thống |
| Trigger | * Người dùng sử dụng chức năng xóa tài khoản |
| Basic flow | * Hệ thống hiển thị cảnh báo người dùng muốn xóa tài khoản hiện tại. * Người dùng chọn “yes” để xóa tài khoản hiện tại * Hệ thống thông báo xóa tài khoản thành công * Hệ thống yêu cầu người dùng đăng nhập |
| Alternative flow | * Người dụng chọn “no” để hủy việc xóa tài khoản |
| Post conditions | Xóa tài khoản khỏi cơ sở dữ liệu |

|  |  |
| --- | --- |
| ID | UC\_APP\_04 |
| Name | Tạo báo cáo theo mẫu |
| Summary | Chức năng tạo báo cáo mới bằng các mẫu báo cáo có sẵn |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ |
| Trigger | Người dùng sử dụng chức năng tạo báo cáo mới |
| Basic flow | * Hệ thống hiển thị danh sách các mẫu báo cáo có sẵn * Người dùng chọn một mẫu báo cáo muốn tạo * Người dùng nhập vào nội dung của các trường dữ liệu * Người dùng chọn lưu báo cáo. |
| Post conditions | Hiển thị các trường dữ liệu cần thêm vào và cho phép người dùng nhập nội dung vào các trường dữ liệu bằng các chức năng có sẵn  Lưu báo cáo vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| ID | UC\_APP\_04\_01 |
| Name | Nhập dữ liệu bằng giọng nói |
| Summary | Chức năng nhập nội dung vào các trường dữ liệu bằng giọng nói |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ và sử dụng chức năng tạo báo cáo mới |
| Trigger |  |
| Basic flow | * Hệ thống hiển thị thông tin của trường dữ liệu cần thêm vào * Hệ thống nhận giọng nói từ người dùng để chuyển thành văn bản |
| Alternative flow | Hệ thống không thể chuyển giọng nói thành văn bản:   * Không có kết nối mạng * Hệ thống tạm thời bảo trì   Người dùng chọn chức năng sửa dữ liệu để sửa dữ liệu theo cách thủ công |
| Post conditions | Chuyển giọng nói của người dùng thành văn bản và nhập vào trường dữ liệu  Chuyển sang trường dữ liệu tiếp theo cần nhập vào |

|  |  |
| --- | --- |
| ID | UC\_APP\_04\_02 |
| Name | Sửa dữ liệu |
| Summary | Chức năng sửa nội dung của các trường dữ liệu sau khi nhập dữ liệu bằng giọng nói |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ và sử dụng chức năng tạo báo cáo mới |
| Trigger | Người dùng chọn chức năng sửa dữ liệu |
| Basic flow | * Hệ thống hiển thị trường dữ liệu đã được thêm vào bằng chức năng nhập dữ liệu bằng giọng nói * Người dùng có thể sửa lại dữ liệu bằng bàn phím ảo/ bàn phím cơ |
| Post conditions | Sửa dữ liệu thành công và chuyển sang trường dữ liệu tiếp theo cần nhập vào |

|  |  |
| --- | --- |
| ID | UC\_APP\_05 |
| Name | Tạo mẫu báo cáo mới |
| Summary | Chức năng tạo mẫu báo cáo mới |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ |
| Trigger | Người dùng chọn chức năng tạo mẫu báo cáo mới |
| Basic flow | * Hệ thống hiển thị các trường dữ liệu với thông tin trống * Người dùng có thể chỉnh sửa thông tin về trường dữ liệu, thêm hoặc xóa trường dữ liệu |
| Post conditions | Tạo mẫu báo cáo mới thành công và lưu mẫu báo cáo vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| ID | UC\_APP\_06 |
| Name | Xem báo cáo |
| Summary | Chức năng cho phép người dùng xem các báo cáo đã lưu của mình |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ |
| Trigger | Người dùng chọn chức năng xem báo cáo |
| Basic flow | * Hệ thống hiển thị các báo cáo đã lưu của người dùng * Người dùng chọn báo cáo muốn xem |
| Post conditions | Hiển thị báo cáo mà người dùng chọn |

|  |  |
| --- | --- |
| ID | UC\_APP\_06\_01 |
| Name | Sửa báo cáo |
| Summary | Chức năng cho phép người dùng sửa báo cáo của mình |
| Actors | User |
| Precondition | User đã đăng nhập thành công bằng tài khoản hợp lệ và sử dụng chức năng xem báo cáo |
| Trigger | Người dùng chọn chức năng sửa báo cáo |
| Basic flow | * Hệ thống hiển thị các trường dữ liệu của báo cáo dưới dạng có thể thay đổi * Người dùng có thể sửa nội dung của các trường dữ liệu bằng cách dùng giọng nói hoặc sửa thủ công * Người dùng chọn lưu báo cáo |
| Alternative flow | Hệ thống không thể sửa nội dung của trường dữ liệu:   * Không có kết nối mạng * Hệ thống đang bảo trì   Hệ thống không thể lưu báo cáo vào cơ sở dữ liệu:   * Không có kết nối mạng * Hệ thống đang bảo trì |
| Post conditions | Sửa báo cáo và lưu vào cơ sở dữ liệu |

* + 1. Đặc tả Use Case
  1. Class Diagram

1. Tổng kết
   1. Các kết quả đã đạt được
   2. Hướng phát triển

Tài liệu tham khảo